**이름 –** 이상호

**학번 –** 12212312

**학과 –** 경영학과

**데이터 –** [Default of Credit Card Clients Dataset | Kaggle](https://www.kaggle.com/datasets/uciml/default-of-credit-card-clients-dataset)

**분석 목표 –** 연령, 성별, 교육 수준, 결혼 여부에 따라 신용카드 사용액, 상환 여부, 한도 등이 어떻게 달라지는 지를 분석하고자 한다.

**분석 내용 –** 이 데이터는 대만의 어떤 신용카드사에서 고객들의 신용카드 사용 정보, 신상 정보들을 모아놓은 것으로 2005년 4월부터 9월까지의 데이터가 존재한다. 이 데이터는 다음과 같은 변수로 이루어져 있다.

ID: ID of each client

LIMIT\_BAL: Amount of given credit in NT dollars (includes individual and family/supplementary credit)

SEX: Gender (1=male, 2=female)

EDUCATION: (1=graduate school, 2=university, 3=high school, 4=others, 5=unknown, 6=unknown)

MARRIAGE: Marital status (1=married, 2=single, 3=others)

AGE: Age in years

PAY\_0: Repayment status in September 2005 (-1=pay duly, 1=payment delay for one month, 2=payment delay for two months, … 8=payment delay for eight months, 9=payment delay for nine months and above)

PAY\_2: Repayment status in August 2005 (scale same as above)

PAY\_3: Repayment status in July 2005 (scale same as above)

PAY\_4: Repayment status in June 2005 (scale same as above)

PAY\_5: Repayment status in May 2005 (scale same as above)

PAY\_6: Repayment status in April 2005 (scale same as above)

BILL\_AMT1: Amount of bill statement in September 2005 (NT dollar)

BILL\_AMT2: Amount of bill statement in August 2005 (NT dollar)

BILL\_AMT3: Amount of bill statement in July 2005 (NT dollar)

BILL\_AMT4: Amount of bill statement in June 2005 (NT dollar)

BILL\_AMT5: Amount of bill statement in May 2005 (NT dollar)

BILL\_AMT6: Amount of bill statement in April 2005 (NT dollar)

PAY\_AMT1: Amount of previous payment in September 2005 (NT dollar)

PAY\_AMT2: Amount of previous payment in August 2005 (NT dollar)

PAY\_AMT3: Amount of previous payment in July 2005 (NT dollar)

PAY\_AMT4: Amount of previous payment in June 2005 (NT dollar)

PAY\_AMT5: Amount of previous payment in May 2005 (NT dollar)

PAY\_AMT6: Amount of previous payment in April 2005 (NT dollar)

default.payment.next.month: Default payment (1=yes, 0=no)

우선 모든 변수에 값이 존재하므로 결측치를 채울 필요는 없다.

print(dataframe.isnull().sum())

>>> LIMIT\_BAL 0

SEX 0

EDUCATION 0

MARRIAGE 0

AGE 0

PAY\_0 0

PAY\_2 0

PAY\_3 0

PAY\_4 0

PAY\_5 0

PAY\_6 0

BILL\_AMT1 0

BILL\_AMT2 0

BILL\_AMT3 0

BILL\_AMT4 0

BILL\_AMT5 0

BILL\_AMT6 0

PAY\_AMT1 0

PAY\_AMT2 0

PAY\_AMT3 0

PAY\_AMT4 0

PAY\_AMT5 0

PAY\_AMT6 0

default.payment.next.month 0

그리고 시각화를 위해서는 변수 이름을 알아보기 쉽게 만드는 것이 좋다. 우선 해당 월의 카드 대금 상환 현황을 나타내는 PAY\_ 변수의 이름을 REPAYMENT STATUS\_ 로 바꾼다.

dataframe.rename(columns = {'PAY\_0' : 'REPAYMENT STATUS\_SEP', 'PAY\_2' : 'REPAYMENT STATUS\_AUG', 'PAY\_3' : 'REPAYMENT STATUS\_JUL'  
 , 'PAY\_4' : 'REPAYMENT STATUS\_JUN', 'PAY\_5' : 'REPAYMENT STATUS\_MAY', 'PAY\_6' : 'REPAYMENT STATUS\_APR'}, inplace = True)

해당 월의 신용카드 청구액을 나타내는 BILL\_AMT 변수의 이름은 CARD BILL\_ 로 바꾼다.

dataframe.rename(columns = {'BILL\_AMT1' : 'CARD BILL\_SEP', 'BILL\_AMT2' : 'CARD BILL\_AUG', 'BILL\_AMT3' : 'CARD BILL\_JUL'  
 , 'BILL\_AMT4' : 'CARD BILL\_JUN', 'BILL\_AMT5' : 'CARD BILL\_MAY', 'BILL\_AMT6' : 'CARD BILL\_APR'}, inplace = True)

해당 월의 선불결제 이용액을 나타내는 PAY\_AMT 변수의 이름을 PREPAID\_ 로 바꾼다.

dataframe.rename(columns = {'PAY\_AMT1' : 'PREPAID\_SEP', 'PAY\_AMT2' : 'PREPAID\_AUG', 'PAY\_AMT3' : 'PREPAID\_JUL'  
 , 'PAY\_AMT4' : 'PREPAID\_JUN', 'PAY\_AMT5' : 'PREPAID\_MAY', 'PAY\_AMT6' : 'PREPAID\_APR'}, inplace = True)

마지막으로 고객의 신용한도를 나타내는 LIMIT\_BAL변수의 이름을 CREDIT LIMIT으로, 고객의 현재 체납 여부를 나타내는 default.payment.next.month 변수의 이름을 DEFAULT 로 바꾼다.

dataframe.rename(columns = {'LIMIT\_BAL' : 'CREDIT LIMIT', 'default.payment.next.month' : 'DEFAULT'}, inplace = True)

또한 SEX, EDUCATION, MARRIAGE, REPAYMENT STATUS\_ 등은 숫자로 표시되어있지만 사실은 범주 변수이므로 그에 맞게 관측값의 이름도 바꾼다.

dataframe.replace({'SEX' : 1}, 'Male', inplace = True)  
dataframe.replace({'SEX' : 2}, 'Female', inplace = True)  
dataframe.replace({'EDUCATION' : 1}, 'Graduate', inplace = True)  
dataframe.replace({'EDUCATION' : 2}, 'Undergraduate', inplace = True)  
dataframe.replace({'EDUCATION' : 3}, 'High School', inplace = True)  
dataframe.replace({'EDUCATION' : 4}, 'Less than Middle School', inplace = True)  
dataframe.replace({'EDUCATION' : 5}, 'Unknown', inplace = True)  
dataframe.replace({'EDUCATION' : 6}, 'Unknown', inplace = True)  
dataframe.replace({'MARRIAGE' : 1}, 'Married', inplace = True)  
dataframe.replace({'MARRIAGE' : 2}, 'Single', inplace = True)  
dataframe.replace({'MARRIAGE' : 3}, 'Others', inplace = True)  
RepayStatVariable = ['REPAYMENT STATUS\_SEP', 'REPAYMENT STATUS\_AUG', 'REPAYMENT STATUS\_JUL', 'REPAYMENT STATUS\_JUN', 'REPAYMENT STATUS\_MAY', 'REPAYMENT STATUS\_APR']  
for i in RepayStatVariable:  
 dataframe.replace({i : -1}, 'No Overdue', inplace=True)  
 dataframe.replace({i : 1}, 'Short-term Overdue', inplace = True)  
 dataframe.replace({i : 2}, '2 Months ~ 6 Months Overdue', inplace = True)  
 dataframe.replace({i : 3}, '2 Months ~ 6 Months Overdue', inplace = True)  
 dataframe.replace({i : 4}, '2 Months ~ 6 Months Overdue', inplace = True)  
 dataframe.replace({i : 5}, '2 Months ~ 6 Months Overdue', inplace = True)  
 dataframe.replace({i : 6}, 'Long-term Overdue', inplace = True)  
 dataframe.replace({i : 7}, 'Long-term Overdue', inplace=True)  
 dataframe.replace({i : 8}, 'Long-term Overdue', inplace=True)  
 dataframe.replace({i : 9}, 'Long-term Overdue', inplace=True)

dataframe.replace({'DEFAULT': 0}, 'Yes', inplace=True)  
dataframe.replace({'DEFAULT': 1}, 'No', inplace=True)

학력을 알 수 없는 고객과 결혼상태가 Others인 고객은 분석에 적합하지 않으므로 삭제한다.

또 관측값이 0으로 되어있는 경우에는 사실상 결측치이므로 삭제한다.

UnkownCustomer = dataframe.loc[(dataframe['EDUCATION'] == 'Unknown') | (dataframe['EDUCATION'] == 0) | (dataframe['MARRIAGE'] == 'Others') | (dataframe['MARRIAGE'] == 0)].index  
dataframe.drop(UnkownCustomer, inplace = True)

REPAYMENT STATUS\_ 에서 0과 -2는 데이터 범례에서 확인할 수 없었다. 보통 연체를 하지 않는 이용자가 대부분인 경우가 많은 점, 0개월 연체나 -2개월 연체는 불가능 하다는 점을 고려해 둘다 모두 연체가 없다고 기록한다.

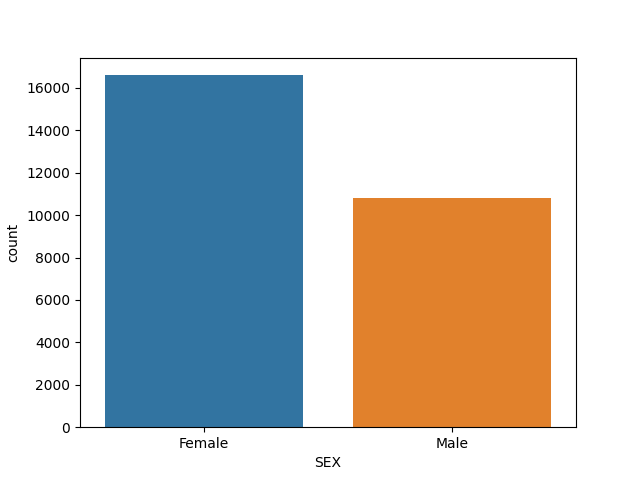
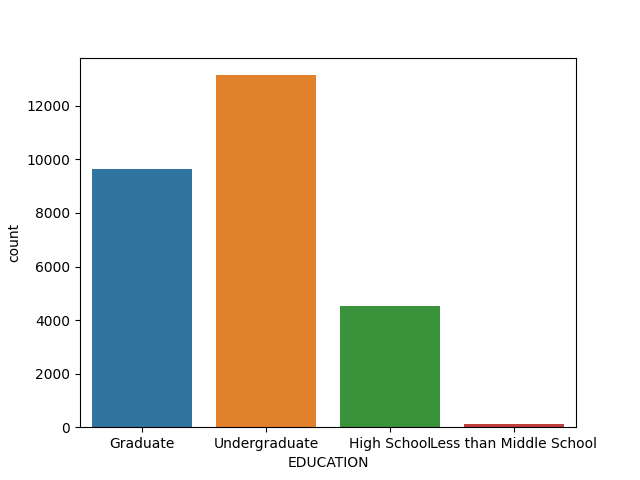
RepayStatVariable = ['REPAYMENT STATUS\_SEP', 'REPAYMENT STATUS\_AUG', 'REPAYMENT STATUS\_JUL', 'REPAYMENT STATUS\_JUN', 'REPAYMENT STATUS\_MAY', 'REPAYMENT STATUS\_APR']  
for i in RepayStatVariable:  
 dataframe.replace({i : 0}, 'No Overdue', inplace=True)  
 dataframe.replace({i : -2}, 'No Overdue', inplace = True)

CARD BILL\_ 이 0 미만인 경우는 상식적으로 설명이 불가능하므로 해당 고객은 삭제한다.

MinersBillCustomer = dataframe.loc[(dataframe['CARD BILL\_SEP'] < 0) | (dataframe['CARD BILL\_AUG'] < 0) | (dataframe['CARD BILL\_JUL'] < 0) |  
 (dataframe['CARD BILL\_JUN'] < 0) | (dataframe['CARD BILL\_MAY'] < 0) | (dataframe['CARD BILL\_APR'] < 0)].index  
dataframe.drop(MinersBillCustomer, inplace = True)

이상으로 데이터 전처리는 끝났다. 이제 어떤 신용카드사에서 고객의 인구통계학적 정보에 따라 신용카드 청구액, 선불결제 이용액, 대금 상환 현황, 채무불이행 가능성 등이 어떻게 다 른지를 분석해본다고 가정하고 데이터를 시각화 해보자

우선 표본의 특성을 간단하게 알아보자.



print(dataframe['SEX'].value\_counts())

>>> Female 16599

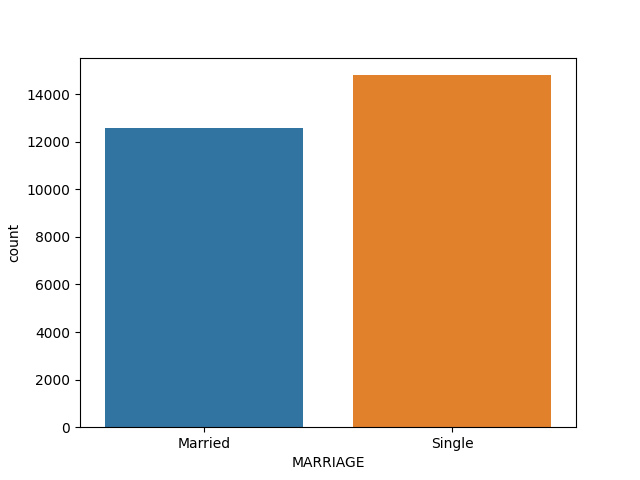
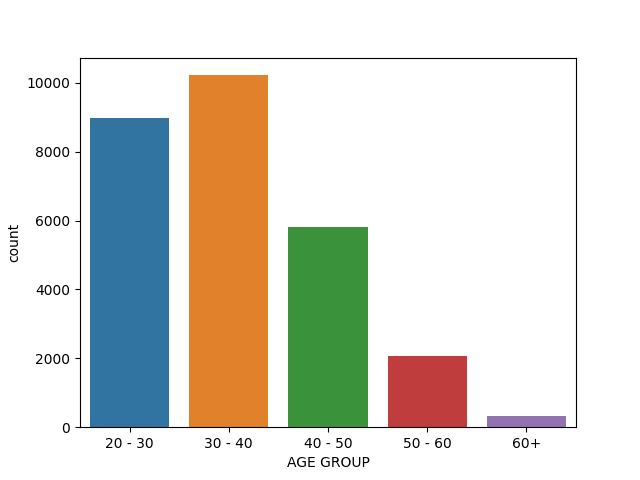
Male 10798

print(dataframe['EDUCATION'].value\_counts())

>>> Undergraduate 13144 Less than Middle School 107

Graduate 9638 High School 4508

성비는 여성이 약 60.5% 남성이 약 39.5%로 여성이 많았다. 또 학력에 있어서는 대학생이 약 48%로 가장 많았고, 대졸 약 35%, 고졸 약 16% 순으로 중졸 이하는 거의 없었다.

print(dataframe['MARRIAGE'].value\_counts())

>>> Single 14798

Married 12599

dataframe.loc[dataframe[(dataframe['AGE'] >= 20) & (dataframe['AGE'] < 30)].index, 'AGE GROUP'] = '20 - 30'  
dataframe.loc[dataframe[(dataframe['AGE'] >= 30) & (dataframe['AGE'] < 40)].index, 'AGE GROUP'] = '30 - 40'  
dataframe.loc[dataframe[(dataframe['AGE'] >= 40) & (dataframe['AGE'] < 50)].index, 'AGE GROUP'] = '40 - 50'  
dataframe.loc[dataframe[(dataframe['AGE'] >= 50) & (dataframe['AGE'] < 60)].index, 'AGE GROUP'] = '50 - 60'  
dataframe.loc[dataframe[dataframe['AGE'] >= 60].index, 'AGE GROUP'] = '60+'

print(dataframe['AGE GROUP'].value\_counts())

>>> 20 - 30 8982

30 - 40 10219

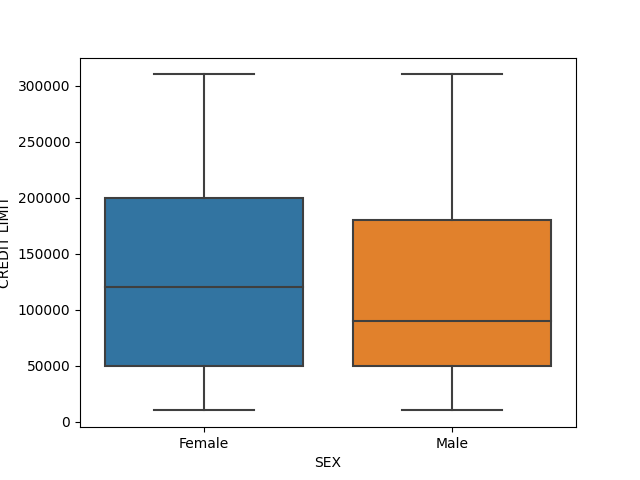
40 - 50 5812

50 - 60 2072

60+ 312

결혼 여부는 미혼 약 54%, 기혼 약 46%로 미혼이 약간 더 많았다. 나이는 범주 변수인 나이 대 변수로 바꾸어 시각화했다. 그 결과 20대 약 32%, 30대 약 37%, 40대 약 21%, 50대 약 7%로 20대와 30대가 가장 많았고 나이대가 올라갈수록 비율이 줄어들었다.

인구통계학적 변수에는 성별, 학력, 결혼여부, 나이대의 4개의 변수가 존재한다. 먼저 성별을 기준으로 변수들을 분석해보자



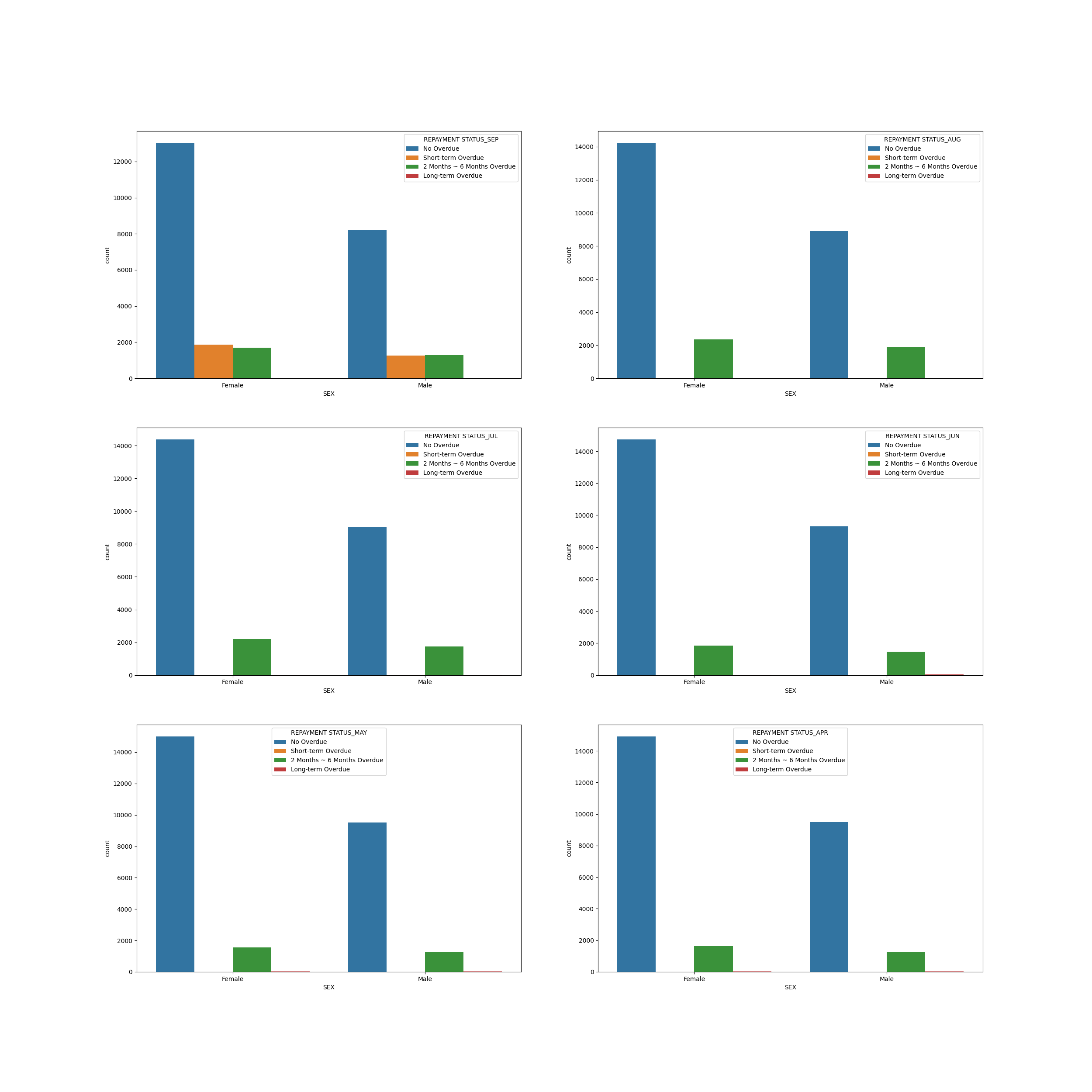
Q1\_cl = dataframe['CREDIT LIMIT'].quantile(0.25)  
Q3\_cl = dataframe['CREDIT LIMIT'].quantile(0.75)  
IQR\_cl = Q3\_cl - Q1\_cl  
dataframe\_CL = dataframe.loc[(dataframe['CREDIT LIMIT'] > Q1\_cl - 0.5 \* IQR\_cl) & (dataframe['CREDIT LIMIT'] < Q3\_cl + 0.5 \* IQR\_cl)]

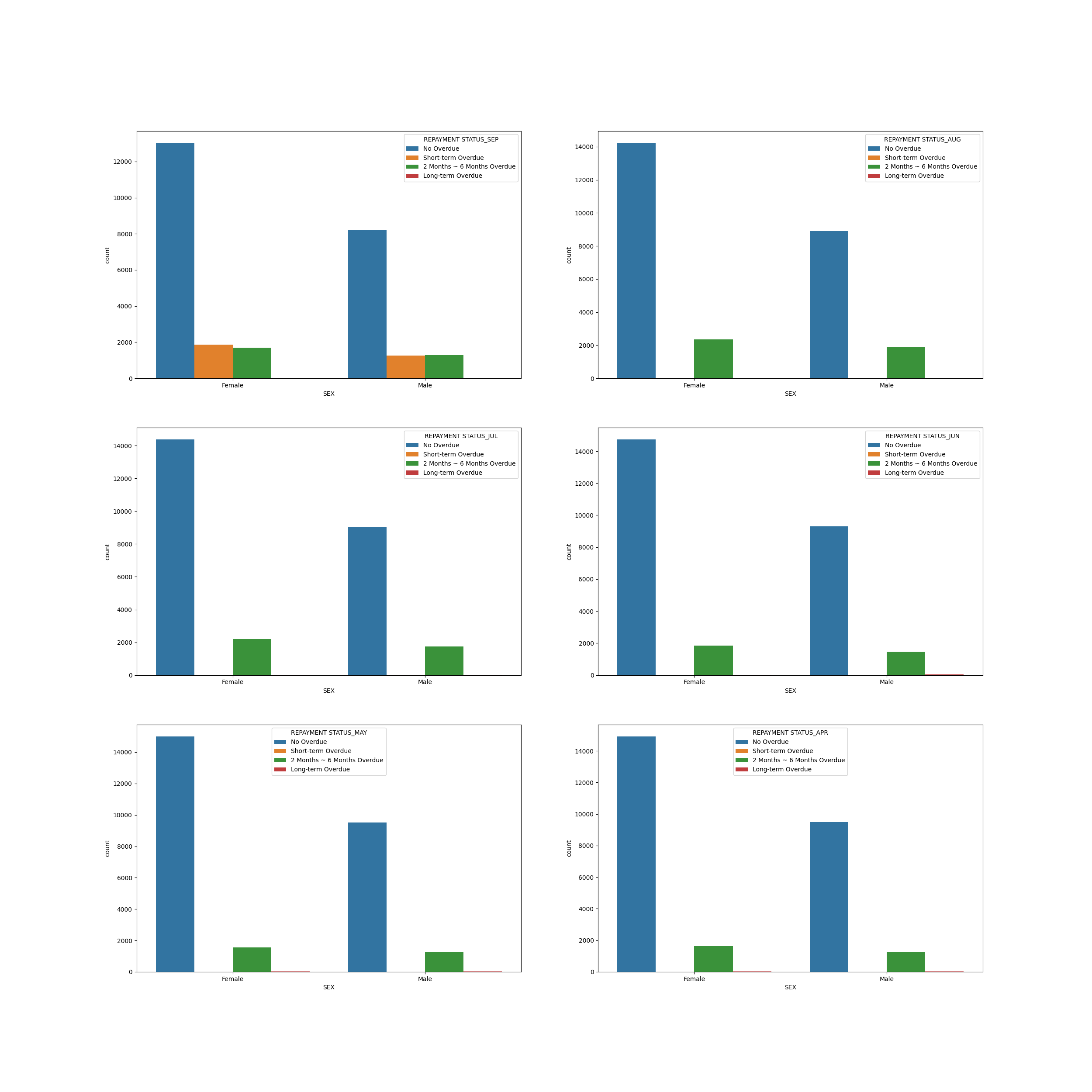
print(dataframe.groupby(['SEX'])['CREDIT LIMIT'].median())

>>> Female 140000.0

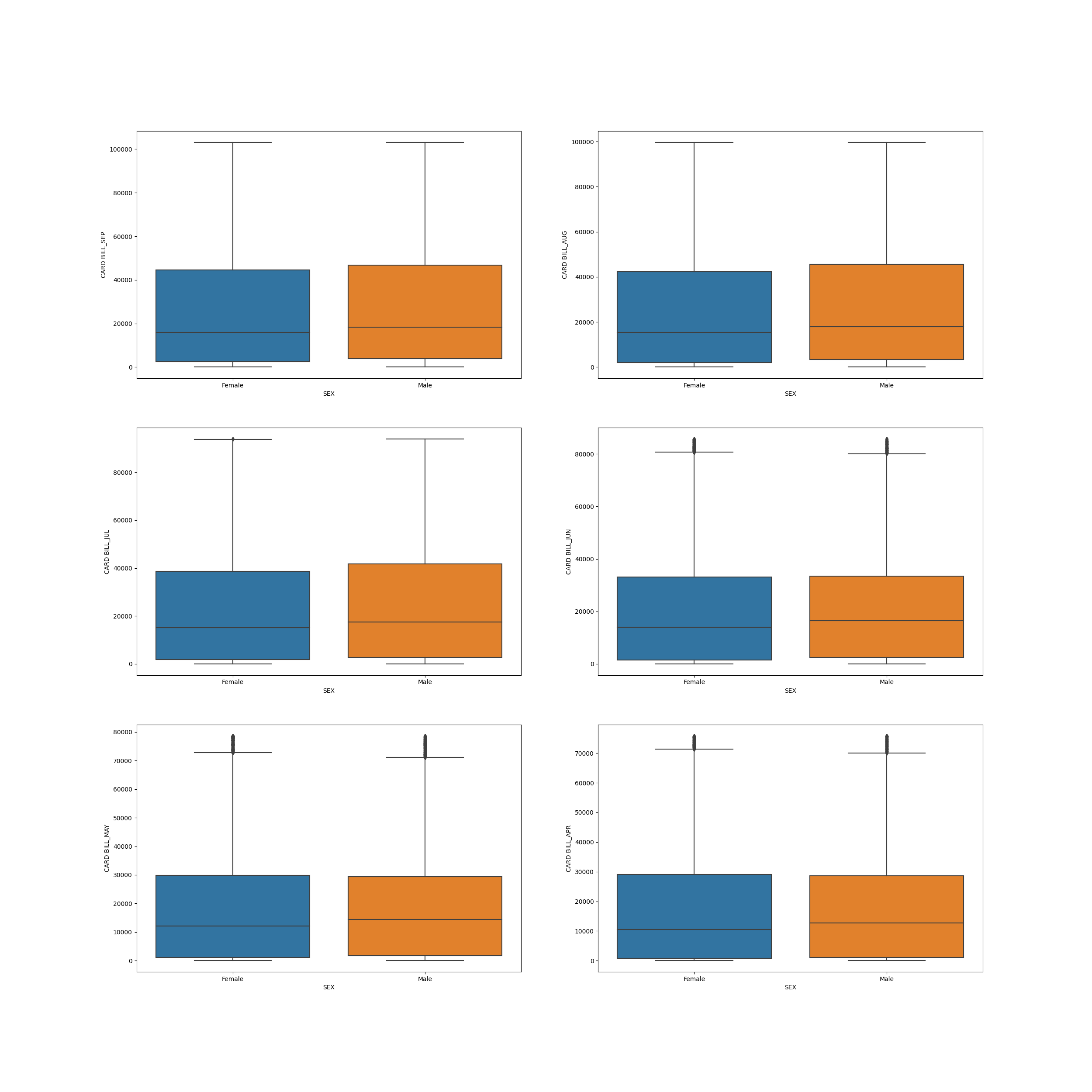
Male 120000.0

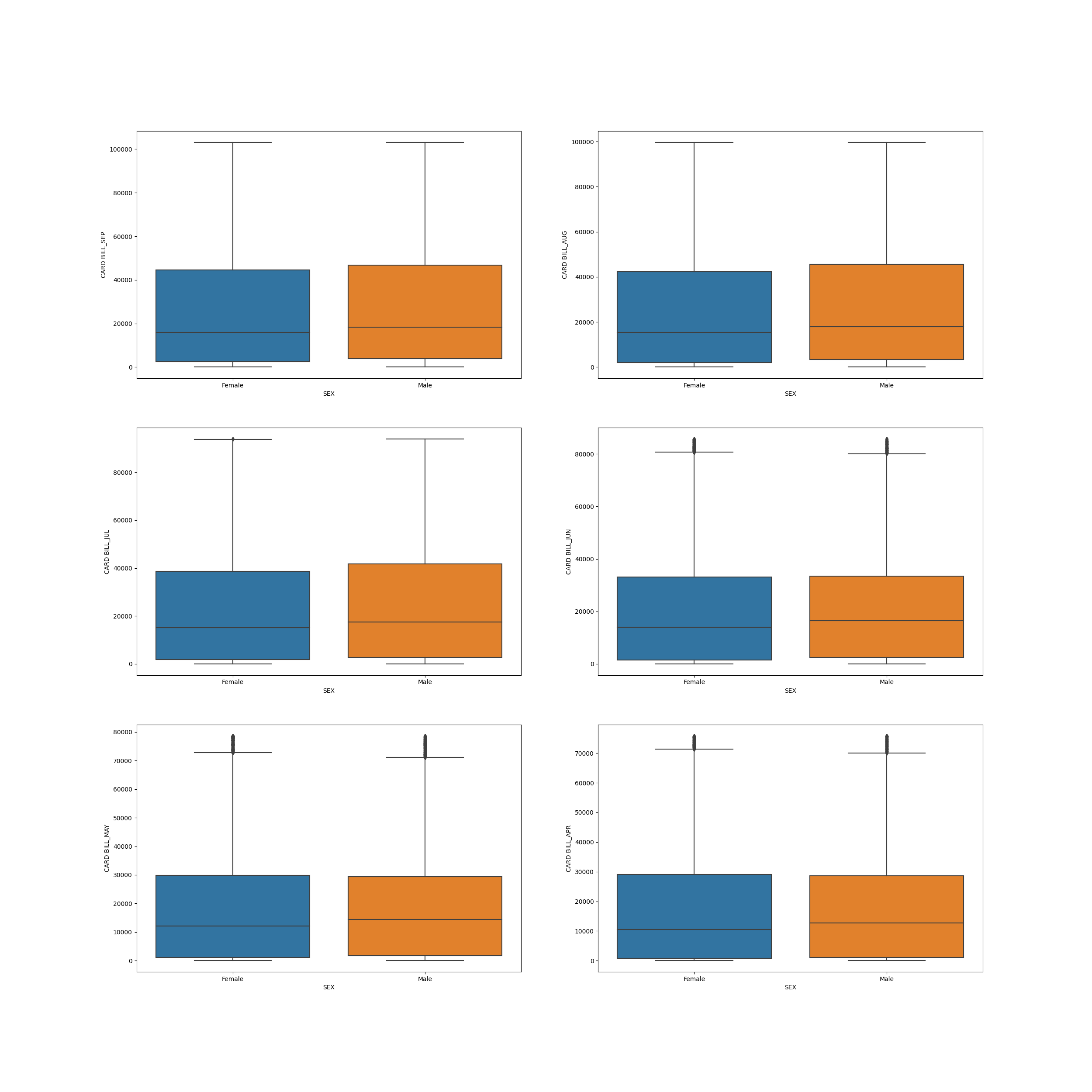
신용한도 변수에 처리를 하지 않은 상태에서는 이상치가 매우 많았다. 따라서 이상치가 보이 지 않도록 관측값의 구간을 조절했다. 이때 신용한도 중앙값은 여성이 2만 신대만달러만큼 즉, 약 16.6% 만큼 높았다.





상환 현황에 있어서는 성별에 따라 차이가 크지 않았다. 비율상의 차이는 약간 있으나 남녀 불문 연체가 없는 고객이 다수였으며 최대 약 13% ~ 15% 정도의 고객은 2개월에서 6개월 정도의 연체가 있었다. 1개월 미만의 단기연체나 6개월 이상의 장기연체의 경우는 9월을 제 외하면 거의 없었다.





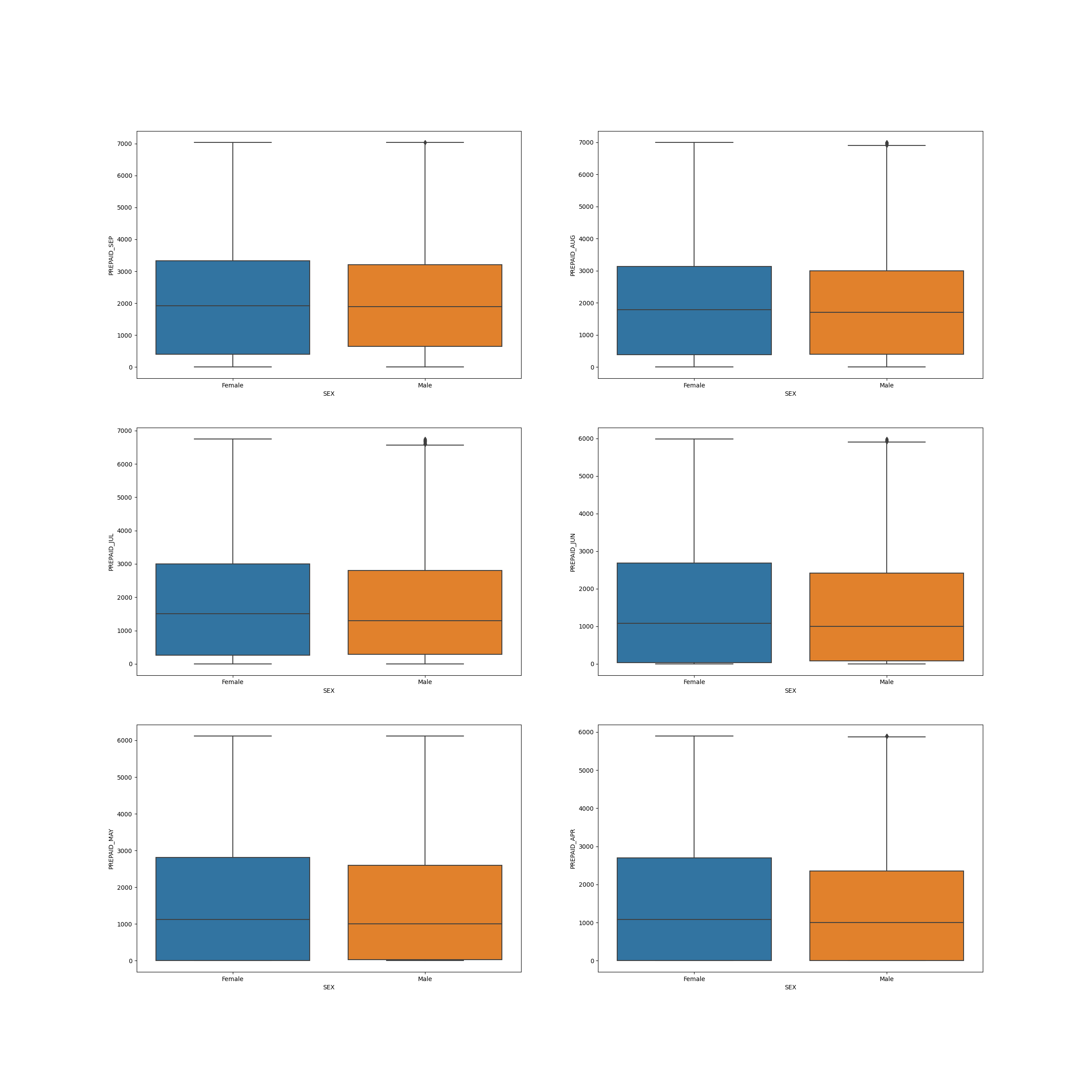
CardBillVariable = ['CARD BILL\_SEP', 'CARD BILL\_AUG', 'CARD BILL\_JUL', 'CARD BILL\_JUN', 'CARD BILL\_MAY', 'CARD BILL\_APR']   
for i in range(len(CardBillVariable)):  
 Q1\_cb = dataframe[CardBillVariable[i]].quantile(0.25)  
 Q3\_cb = dataframe[CardBillVariable[i]].quantile(0.75)  
 IQR\_cb = Q3\_cb - Q1\_cb  
 dataframe\_CB = dataframe.loc[(dataframe[CardBillVariable[i]] > Q1\_cb - 0.5 \* IQR\_cb) & (dataframe[CardBillVariable[i]] < Q3\_cb + 0.5 \* IQR\_cb)]

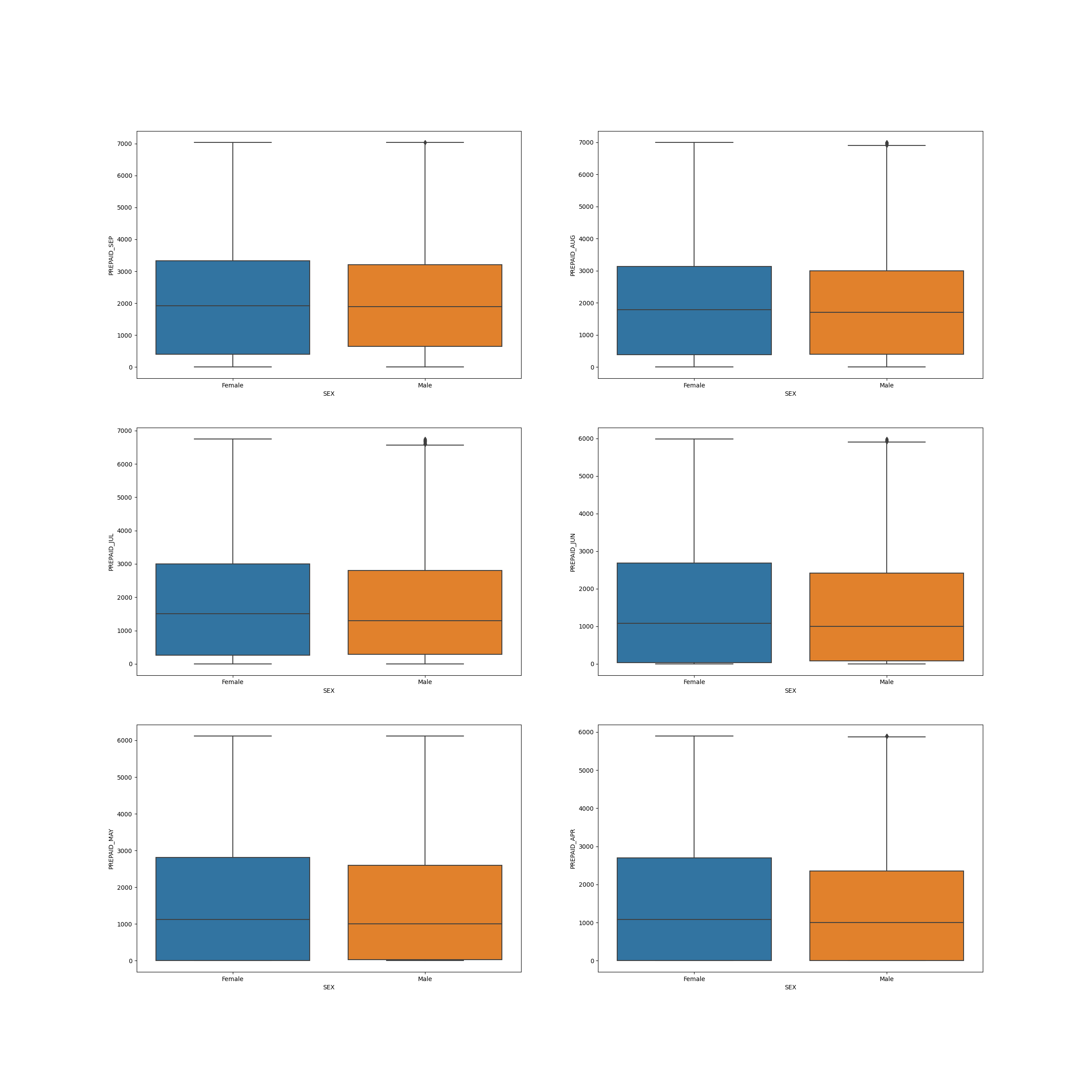
print(dataframe.groupby(['SEX'])[CardBillVariable[i]].median())

>>> Female 24110.0 Male 27284.0 Name: CARD BILL\_SEP, dtype: float64

Female 23201.0 Male 26015.5 Name: CARD BILL\_AUG, dtype: float64 Female 21865.0 Male 23610.0 Name: CARD BILL\_JUL, dtype: float64 Female 20204.0 Male 20245.0 Name: CARD BILL\_JUN, dtype: float64 Female 19225.0 Male 19374.0 Name: CARD BILL\_MAY, dtype: float64 Female 18375.0 Male 19066.5 Name: CARD BILL\_APR, dtype: float64

신용카드 청구액 변수도 신용한도 변수와 마찬가지로 관측값의 구간을 제한했다. 이때 신용카드 청구액 중앙값은 남성이 대체로 높은 것으로 나타났다. 남성은 여성보다 평균적으로 최대 약 12%가량 더 많은 금액을 신용카드로 사용했다.



PrepaidVariable = ['PREPAID\_SEP', 'PREPAID\_AUG', 'PREPAID\_JUL', 'PREPAID\_JUN', 'PREPAID\_MAY', 'PREPAID\_APR']  
Fig3 = plt.figure(figsize = (25, 25))  
for i in range(len(PrepaidVariable)):  
 Q1\_pr = dataframe[PrepaidVariable[i]].quantile(0.25)  
 Q3\_pr = dataframe[PrepaidVariable[i]].quantile(0.75)  
 IQR\_pr = Q3\_pr - Q1\_pr  
 dataframe\_PR = dataframe.loc[(dataframe[PrepaidVariable[i]] > Q1\_pr - 0.5 \* IQR\_pr) & (dataframe[PrepaidVariable[i]] < Q3\_pr + 0.5 \* IQR\_pr)]

print(dataframe.groupby(['SEX'])[PrepaidVariable[i]].median())

>>> Female 2230.0 Male 2184.0 Name: PREPAID\_SEP, dtype: float64

Female 2107.0 Male 2021.0 Name: PREPAID\_AUG, dtype: float64

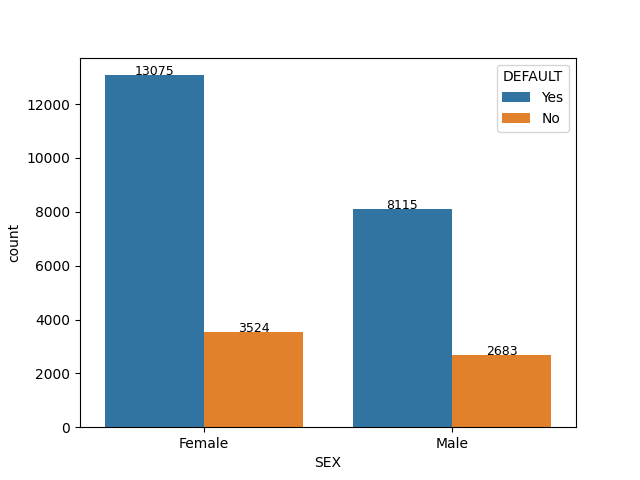
Female 2000.0 Male 1763.0 Name: PREPAID\_JUL, dtype: float64

Female 1650.0 Male 1500.0 Name: PREPAID\_JUN, dtype: float64

Female 1700.0 Male 1507.5 Name: PREPAID\_MAY, dtype: float64

Female 1685.0 Male 1431.5 Name: PREPAID\_APR, dtype: float64

선불결제 이용액 변수도 앞의 경우와 같은 처리를 했다. 이때 중앙값은 남성보다 여성이 대 체로 높았다. 여성은 평균적으로 남성보다 최대 17% 더 많은 금액을 선불결제로 사용했다.



채무 불이행 가능성, 즉 연체가 발생할 가능성은 여성 약 21%, 남성 약 24%로 남성이 약간 더 높았다. 그러나 큰 의미를 부여할만한 차이는 아니다.

정리해보면, 부여받는 신용한도는 여성이 더 높았다. 이때 신용카드 청구액 즉, 신용거래를 한 금액은 남성이 약 10%정도, 선불결제 이용액은 여성이 약 15% 정도 높았다. 채무불이행에 빠질 확률은 남성이 절대적으로 3%p 가량, 상대적으로 14% 가량 높았다. 신용

카드사라면 신용거래를 약간 더 많이하고 그만큼 연체할 확률이 높은 남성에게 상대적으로 약간 적은 한도를 부여하는 것이 합리적이라고 볼 수 있다.

다음으로 학력 변수를 중심으로 데이터를 분석해보자.